

# A Supplemental Information

“Synthetically Generated Text for Supervised Text Analysis”

Andrew Halterman

Michigan State University

ahalterman0@gmail.com

*Political Analysis (accepted November 2024)*

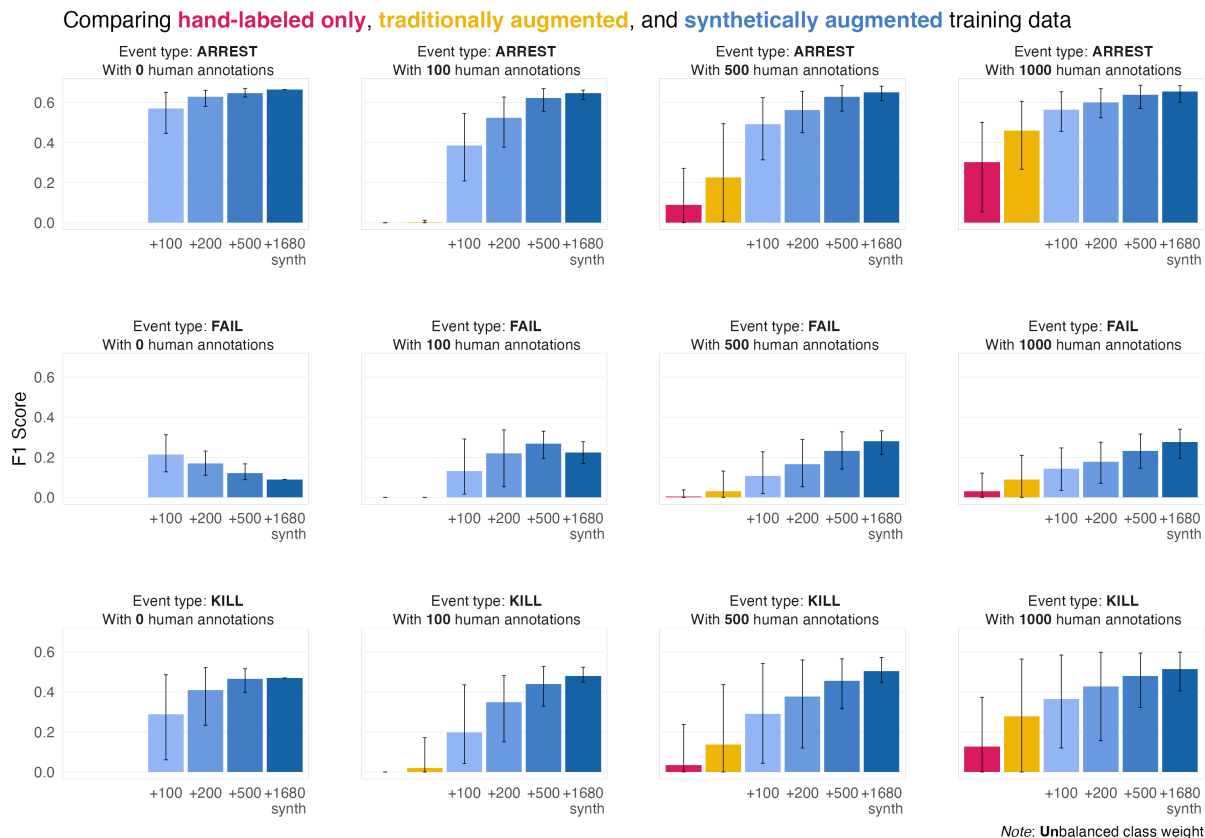
## A.1 Initial headlines for prompting **IndiaPoliceEvents** events

The full set of events in the India Police Events dataset also includes two aggregated event types: FORCE (either ARREST or KILL, along with other uses of police force) and ANY (ARREST, KILL, or any other mention of police events identified by the annotator). I include headlines describing the two aggregated classes to provide negative examples for each class that also discuss police actions. Table SI 1 reports the initial set of headlines.

## A.2 India Police Events classifier extensions

### A.2.1 Traditional augmentation

The traditional data augmentation using in Validation 1 and reported in Figure 2 uses a data augmentation technique described in (Wei and Zou 2019). They propose four transformations to apply to a document: synonym replacement (swap a word with its synonym), random insertion (find a synonym of a word in the text, add in a random position), random swap (randomly swap the position of two words in the text), random deletion (randomly delete a word from the sentence). Following the recommendations in their paper, I transform 10% of words and generate 4 new modified document for each original document. I use a Python implementation of their technique (Morris et al. 2020).



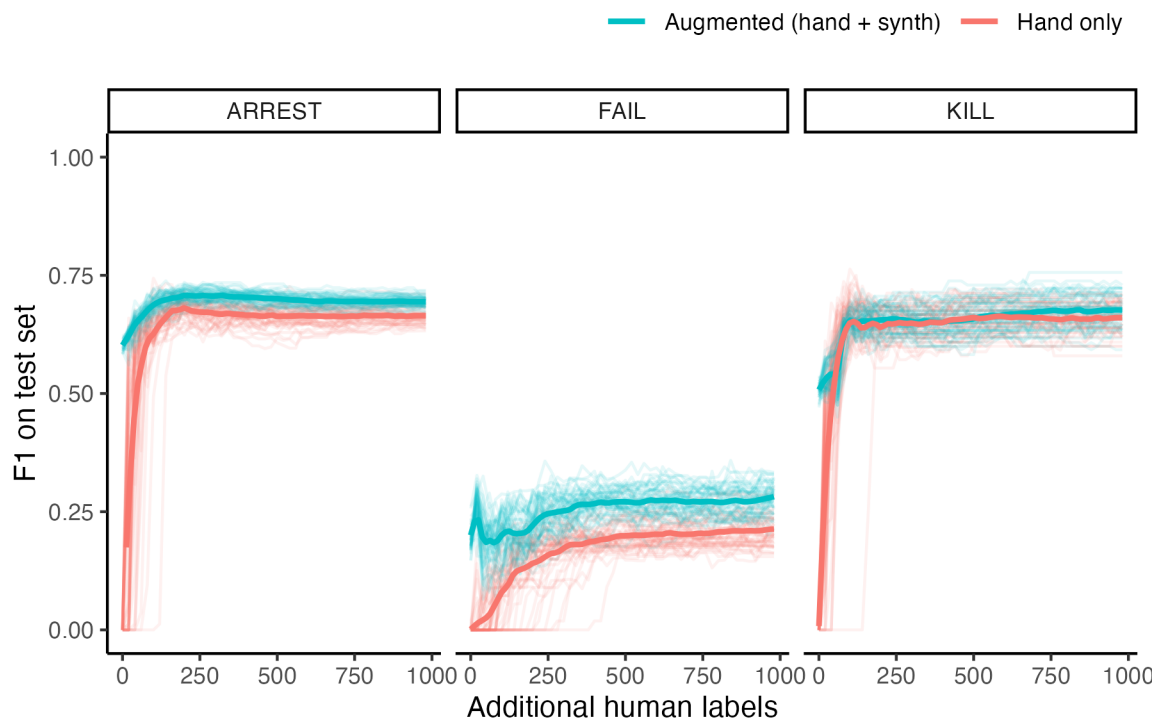
**Figure SI 1:** *Alternative version of main text Figure 2 without class weights. Mean F1 performance on an evaluation set for three classes (ARREST, FAIL, KILL) with increasing sizes of the hand-annotated training set (100, 500, 1,000). The red bar shows performance with hand-labeled data only. The orange bar shows performance with traditional data augmentation (Wei and Zou 2019). The four blue columns show the performance with different numbers of synthetic training examples added to the hand-labeled data.*

## A.2.2 Class imbalance

The classes in the India Police Events dataset are highly imbalanced, with some events appearing in fewer than 0.5% of documents. The main figure (Figure 2) reports results using balanced class weights in the logistic regression classifier. When computing the loss function, examples are weighted to give higher weight to mispredictions on the rare class. Figure SI 1 shows results when using an unweighted logistic regression classifier. Here, the improvements from synthetic augmentation are much more dramatic. Because the synthetic text is more balanced than the original text, using it as augmentation helps address the class imbalance problem without the standard class weights loss adjustment.

### A.2.3 Active learning

A common recommendation when annotating rare documents is to use active learning, where the classifier is used during the annotation process to select the most informative document for hand labeling (Miller, Linder, and Mebane 2020). Figure SI 2 shows the results of an active learning annotation process with synthetic data. I simulate a process of hand annotation by sampling sentences and their labels from the IndiaPoliceEvents dataset to create a training set. It begins with a set of 10 labeled real documents, including one positive document for the class. I then use uncertainty-based active learning to select the most informative documents for “annotation” (or, in this case, to use the label provided by the dataset). Each annotation is a random draw from the set of individual annotator answers for each document. Active learning helps address the class imbalance problem and provides a very strong baseline model. Augmenting with synthetic data provides modestly higher performance, or, alternatively, similar accuracy with fewer hand annotations.



**Figure SI 2:** *Active learning classification results. Each annotation is a random draw from the set of individual annotator answer for each document. The performance is slightly lower overall, and the best performance is achieved with synthetic augmentation and a relatively small number of hand annotations (fewer than 100).*

<b>Event Type</b>	<b>Hand-Written Title for Prompting</b>
KILL	Three killed in police firing during communal violence
KILL	Police kill three protesters during Gujarat unrest
KILL	Two killed after police open fire on Gujarat crowd
KILL	One killed, many injured in police firing
ARREST	Two arrested after communal violence
ARREST	Police detain scores in Gujarat violence
ARREST	Dozens detained after night of communal unrest
ARREST	Police announce arrest of protest leaders
ARREST	Police book four for inciting violence
ARREST	Police round up activists in Gujarat
ARREST	Police arrest 20 in Gujarat unrest
FAIL	Police stand by helplessly as communal violence breaks out
FAIL	Police fail to respond to violence
FAIL	Police fail to step in, allowing violence to spread
FORCE	Police lathi-charge crowd in Gujarat
FORCE	Police beat protestors
NONE	Police announce new hiring initiative
NONE	Local assembly holds hearing on police budget
NONE	Four killed in industrial accident
NONE	Six killed in traffic accident
NONE	Four dead after building collapse
NONE	BJP forms new government in Gujarat
NONE	Electricity prices rise in Gujarat
NONE	Agriculture minister announces new subsidies
NONE	Court rules on Gujarat land dispute
NONE	Local assembly holds hearing on police budget
NONE	Commissioner congratulates police academy class
NONE	New hospital opens in Gujarat
NONE	Gujarat student wins national spelling bee
NONE	Gujarat wins national cricket championship

**Table SI 1:** *Hand-written headlines used to prompt GPT-2 to generate synthetic text for the India Police Events dataset.*

## B India Police Events end-to-end example

Here, I provide a more detailed walk-through of the process of generating synthetic text to train a supervised classifier. Specifically, I expand on Validation 1 using synthetic news text to train a classifier to identify police responses to communal violence in India.

The India Police Events dataset (Halterman et al. 2021) uses five labels: whether police made an **ARREST**, **KILLED** someone, or **FAILED** to act, along with two composite labels: any use of **FORCE** (either **ARREST** or **KILL**, along with non-lethal violence against civilians), and a label **ANY ACTION** for any mention of police action (excluding **FAIL** and including other police actions not covered by the other labels).

Our goal is to train a classifier to identify these labels in text, using little to no hand-labeled data. (Though note that some hand-labeled data is required to evaluate the performance of the model.) For each label, we want a set of stories that contain that event to serve as positive cases, along with a set of negative stories that do not contain the event. The success of the process will depend on reliably generating synthetic articles that reflect the label.

### B.1 Generating text

#### B.1.1 Picking a generation technique

Step 1 of the process is to select the method for generating text. We have a choice between prompting and adaptation, with the possible option of changing the sampling parameters. News text is straightforward to prompt: it is well represented in the pretraining corpus of almost any LLM and the content of a news story can be easily elicited by a headline. Adapting a model to generate text is more technically challenging, and requires a reference corpus. To use adaptation to generate stories about a specific event (e.g., from the **ARREST** class) would require adapting/fine-tuning the model on a set of stories that describe arrests.

Because we do not have this corpus, adaptation does not make sense here, and we can proceed with the (easier) prompting approach.

### B.1.2 Writing prompts

Writing the prompts, which take the form of headlines in this situation, involves using pre-existing substantive knowledge to elicit stories with the desired class labels.

For example, to generate a positive synthetic example for the KILL class, we can write the headline:

*“Two killed after police open fire on Gujarat crowd”*

We can also incorporate contextual knowledge from the case. For example, Indian news stories often use the noun phrase “police firing” when describing police using firearms:

*“One killed, many injured in police firing in Gujarat”.*

Similarly, we can incorporate case-specific descriptions of events, such as the Indian police’s use of lathis (batons) as a crowd control tactic, or the term “first information report (FIR)”, referencing the police opening a case following a public complaint:

- *“Police lathi-charge crowd in Gujarat”*
- *“Police fail to initiate FIR against rioters in Gujarat”*

Finally, to further guide the model, we can provide context about the news source, the country, and a reminder of the desired class, alongside the headlines described above. For example:

*“The following is a story from the Times of India. It describes police arresting people. Police detain scores in Gujarat violence (March 3, 2002)”*

## B.2 Generating text and varying sampling parameters

The best generation hyperparameters in this case cannot be selected automatically using the adversarial method. Because the events of interest are very rare in the corpus, synthetic text that looks very similar to the real text might not include any of the positive examples we are looking for. Instead, I select standard generation hyperparameters. These parameters could be tweaked following manual reading of the synthetic text, relying on a qualitative judgement about text quality. Here, I use `gpt-2-xl` running locally, with `top_k=10`, `top_p=0.9`, and `temperature=0.9`.

## B.3 Evaluating the quality of synthetic text

Our synthetic text should cover the semantic space of our real text. One simple way to verify this is by plotting the embeddings of the real and synthetic text in two dimensional space. I embed all the text, real and synthetic, using a sentence transformer model (Reimers and Gurevych 2019). Specifically, I use `all-mpnet-base-v2` model, which has a longer context window and higher reported performance than most sentence transformer models. I then transform the embeddings into a 2D representation using PCA. Because we're interested in how synthetic text covers the real text, I fit PCA using only the embedded real text, then apply the same transformation to the synthetic text. Plotting both allows a quick, heuristic visual check of the synthetic text's coverage.

Figure SI 3 reveals areas where the synthetic text does not overlap with the real text. Specifically, real stories describing party politics and discussions of the demolished Babri mosque/Ram temple construction do not have any similar synthetic stories. Adding new headlines to generate synthetic text (e.g., "VHP leaders hold meeting with politicians in Gujarat", "Editorial: pray for the Ram temple in Gujarat") produces new synthetic stories that better cover the semantic space (Figure SI 4)

Figure SI 5 shows the change in evaluation accuracy for classifiers trained on either the

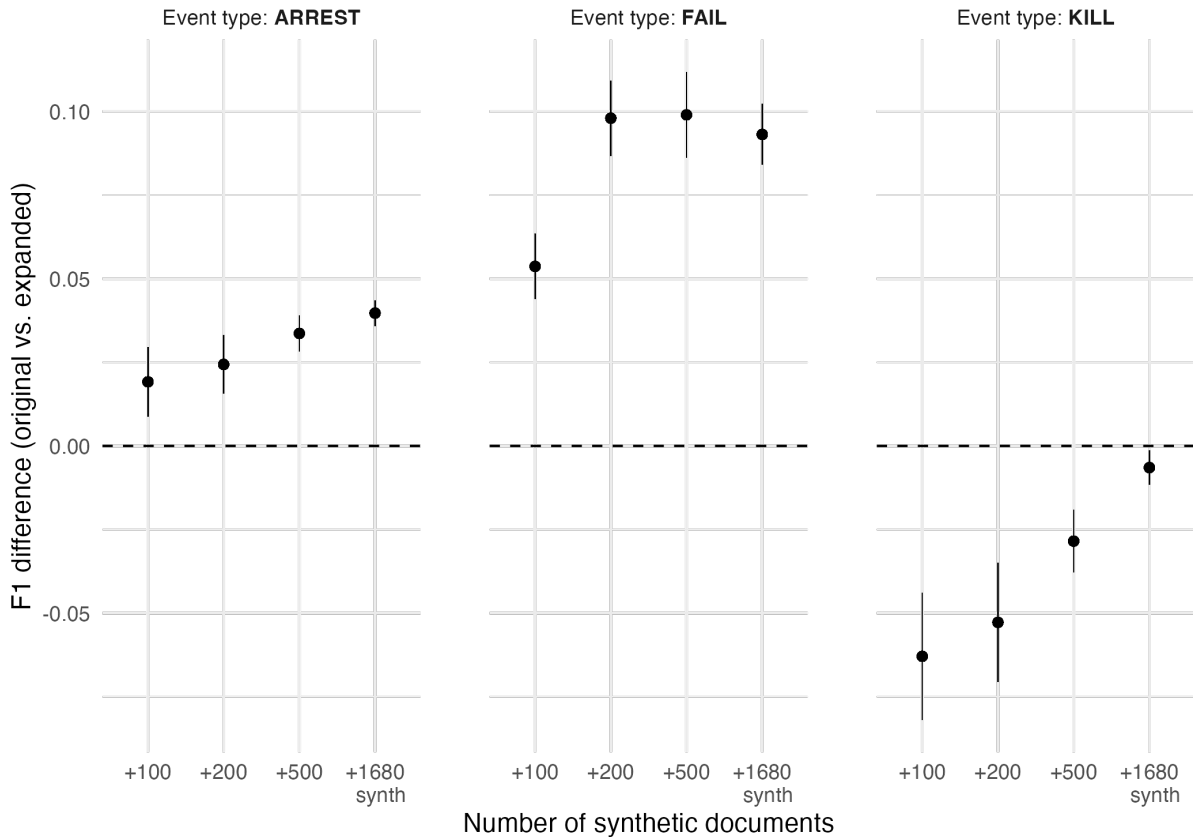




**Figure SI 3:** Comparing real (blue) and synthetic (red) India Police Events text embeddings in 2D PCA space



**Figure SI 4:** After adding new synthetic examples. Comparing real and synthetic India Police Events text embeddings in PCA space



**Figure SI 5:** *Difference in classification performance between original and expanded synthetic examples (with 500 hand annotations). 50 random samples, with 95% confidence interval for a two-tailed t-test.*

original set of synthetic stories or an expanded set of synthetic stories using new headlines. The drop in accuracy for the KILL class is attributable to worsened class imbalance. Only one KILL headline was added, making the category rarer in the synthetic text. In applied work, the composition of the synthetic text can be varied and the performance of the classifier evaluated on a dev set to improve classifier performance.

## B.4 Fitting a classifier

At this point, we have high quality synthetic text and can fit a supervised classifier to predict the labels. Here, I opt for a simple logistic regression classifier, which is fast to train and easy to interpret. As a document representation, I use sentence embeddings produced

by a sentence transformer model (Reimers and Gurevych 2019). These embeddings are fast and simple to compute, but do a good job of representing the semantic content of the text. In an applied setting, a researcher might use a more complex classifier, tune the hyperparameters of the classifier, or tune the document embedding technique to improve performance. If researchers engage in this kind of hyperparameter tuning, they should be careful to follow a standard train/dev/test split to avoid overfitting to the test set.

## C Populism Prompts and Validation

I attempt to provide conceptually clear examples of populist rhetoric by writing prompts that are meant to elicit populist statements from the language model, without conflating populism with other political stances, such as anti-migrant, right-wing, or Eurosceptic positions, or opposition to the party in power.

The classifier is a sentence transformer document embedding model (paraphrase-multilingual-mpnet-base-v2) with a classifier applied to the embeddings. In contrast to the first validation, where the embeddings are frozen, I update the embeddings in this validation using an efficient contrastive learning technique (Tunstall et al. 2022). This classifier involves two steps: first, updating the embeddings for each document so documents with the same label have more similar embeddings, while documents with different labels have their embeddings pushed further apart, reducing their similarity. The model updates through labeled examples, updating the embeddings. Finally, a separate classifier is fit on the updated embeddings. The contrastive learning approach changes the embeddings to make the classification task more straightforward, increasing classification accuracy even with relatively few labeled examples.

As a first evaluation, the model trained on synthetic populist and non-populist text achieves an accuracy of 0.93 and macro F1 score of 0.85 in an internal train/test split of the synthetic data. This indicates that the classifier can reliably distinguish between

synthetic sentences from populist and non-populist prompts, but does not necessarily generalize to the model’s ability to identify populist statements in real text. The main text includes more useful validations.

Table SI 6 provides full populism classifier results on the set of hand-annotated manifesto statements. Because populist statements are relatively rare, I stratify the sample to annotate by predicted populism score to take a sample of the 100 highest scoring sentences, 100 lowest scoring sentences, and 250 randomly selected sentences from middle of the predicted range. A small number of sentences are excluded because they were not relevant (e.g., section titles, word fragments).

Table SI 2 shows an example non-populist prompt using the Manifesto Project codebook. Table SI 3 below show the prompts used to generate non-populist text.

Table SI 4 shows 10 randomly selected sentences from United Kingdom Independence Party manifestos and their predicted populism scores.

Table SI 5 shows results from an apparent false negative—the PopuList dataset (Rooduijn et al. 2019) codes the Northern League in 2013 as populist, but the classifier identifies *no* quasi-sentences with a predicted populism score above 0.5. A complete hand coding of the manifesto reveals that the party’s manifesto does not contain any overtly populist sentences. The table reports sentences that have the sentences that are subjectively most populist, but none of them contain the definition of populism discussed in the main text (Mudde 2004).

Economic (414)	Orthodoxy	<p>A political party is calling for economic orthodoxy, the need for economically healthy government policy making. May include calls for:</p> <ul style="list-style-type: none"> <li>• Reduction of budget deficits;</li> <li>• Retrenchment in crisis;</li> <li>• Thrift and savings in the face of economic hardship;</li> <li>• Support for traditional economic institutions such as stock market and banking system;</li> <li>• Support for strong currency.</li> </ul> <p>Write a list of 10 statements that this party in {country} might make (in the {language} language) in its party platform:</p>
Welfare State (504.0)	State Expansion	<p>A political party supports welfare state expansion. Favourable mentions of need to introduce, maintain or expand any public social service or social security scheme. This includes, for example, government funding of:</p> <ul style="list-style-type: none"> <li>• Health care;</li> <li>• child care;</li> <li>• Elder care and pensions;</li> <li>• Social housing.</li> </ul> <p>Note: This category does NOT include education.</p> <p>Write a list of 10 statements that this party in {country} might make (in the {language} language) in its party platform:</p>

**Table SI 2:** *Example Comparative Manifesto Project prompts for non-populist text. The ten most common codes are 414, 201.0, 416.2, 504.0, 403.0, 703.0, 304.0, 402, 705.0, 502.0. See Volkens et al. (2021) for details.*

“A political party supports existing political institutions, multiculturalism, globalization, and respect for the existing political process. It supports equality for all people and welcomes immigrants.

Write ten statements that this party in {country} might make (in {language}):”

“A party in {country} believes that taxes are too high. It wants to reduce taxes and cut government spending. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} wants to spend more money on schools, housing, and the military. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} believes that the government should spend more money on healthcare, education, and infrastructure. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} wants to raise the minimum wage. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} is calling for greater support for teachers and police. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} is criticizing the foreign policy of its opposing party. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} is criticizing populism as a threat to {country}. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} is criticizing the high rate of unemployment. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} believes the country is on the wrong path. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

“A party in {country} is criticizing the party is is campaigning against. Write 10 statements that {adjective} party might make in the style of a political manifesto (in the {lang} language):”

---

**Table SI 3:** *Hand-written prompts used to generate non-populist text. “adjective” refers to the country adjective such as “Swiss” or “Dutch”.*

Sentence	Score
As a minimum, we will seek continued access on free-trade terms to the EU’s single market.	0.05
A GRAMMAR SCHOOL IN EVERY TOWN The state education system of grammar, secondary modern and technical schools was designed to make a high standard of education available to all, irrespective of social background.	0.06
Climate Change Act, the most expensive piece of legislation in history.	0.06
BRITISH CULTURE	0.05
3.	0.09
In the longer term, we will aim to restore the personal allowance to those earning over £100,000 and make 40 per cent the top rate of tax for all, as it used to be.	0.06
in particular for children, the elderly and people on low incomes.	0.04
We will also exempt foodbanks and charity shops from charges imposed by local authorities to dispose of unwanted food waste and other goods.	0.05
UKIP will restore Britain’s armed forces to their rightful place among the most professional, flexible and effective fighting forces in the world, and we will sign a new military covenant with our brave heroes.	0.12
Prisoners are encouraged to deal with drug addiction problems during their incarceration, and we do them no favours by not taking a tough line.	0.05

**Table SI 4:** *Ten randomly selected UKIP manifesto sentences and their predicted populist scores.*



---

“Far dimagrire lo Stato, i cittadini danno già troppo”

[translated] “Major slimming down of the State: citizens already give too much”

“Incremento della lotta per la legalità, per il contrasto ai fenomeni della immigrazione clandestina”

[translated] “Increase in the fight for legality, to contrast the phenomena of illegal immigration”

“Nuove azioni per favorire la concorrenza nel settore energetico e contrastare gli oligopoli”

[translated] “New actions to encourage competition in the energy sector and counter oligopolies”

“Razionalizzare la distribuzione territoriale degli istituti e degli insegnamenti universitari”

[translated] “Rationalize the territorial distribution of institutes and university teaching”

“Eventuali salvataggi bancari devono essere solo a tutela dei risparmiatori e non degli azionisti di controllo”

[translated] “Any bank bailouts must only protect savers and not controlling shareholders”

“Votare il dimezzamento degli emolumenti dei parlamentari.”

[translated] “Vote to halve the salaries of parliamentarians.”

“Elezione diretta e popolare del Presidente della Repubblica”

[translated] “Direct and popular election of the President of the Republic”

“Dimezzamento dei costi della politica. Abolire il finanziamento pubblico dei partiti (nessun fondo pubblico ai partiti)”

[translated] “Halving of the costs of politics Abolishing public funding of parties (no public funding to parties)”

“Più Europa dei Popoli, meno euro-burocrazia”

[translated] “More Europe of Peoples, less Euro-bureaucracy”

“La difesa e il sostegno alla famiglia, comunità naturale fondata sul matrimonio tra uomo e donna”

[translated] “The defense and support of the family, a natural community founded on marriage between a man and a woman”

---

**Table SI 5:** *Hand validation of populism in Italy’s Northern League 2013 manifesto, showing the sentences that appear most populist in a hand coding of the document. The populism classifier identifies no sentences with a predicted populism score above 0.5, while PopuList codes the party-election as populist. While some of the sentences contain slightly populist statements, none appear to be overtly anti-elite and pro-common person.*

	Precision	Recall	F1	Support
Non-populist	0.91	0.94	0.92	338
Populist	0.66	0.55	0.60	71
Accuracy	0.87	0.87	0.87	
Macro avg	0.78	0.75	0.76	409
Weighted avg	0.87	0.87	0.87	409

**Table SI 6:** *Performance of a classifier trained on synthetic text, evaluated on a new set of hand-annotated English language manifesto sentences.*

## D Fine-tuning Ukraine war synthetic tweets

Existing named entity recognition models do not identify named weapons systems in text. For researchers studying the provision, proliferation, and use of weapons systems, automatically identifying mentions of named weapons systems in open source text provides a valuable source of data. When annotating the tweets to train the model, I label only named weapons systems (e.g, “Bayraktar TB2”), not generic terms (“drone”).

I collected 27,360 tweets from February 2022 to July 2022 from four open source analysts who reported heavily on the initial phases of the war in Ukraine: @markito0171, @oryxspioenkop, @osinttechnical, @uaweapons.

I adapted (continue pretraining on a new domain on the next token prediction objective) GPT-2-large using the Huggingface transformer library using default training parameters. I add the special GPT-2 tokens <|startoftext|> and <|endoftext|> to the tweets and use the pretrained <|pad|> token to bring all tweets to the same length.

I deliberately do not optimize the fine-tuning (adaptation) step, given the time and resource demands. It is unrealistic to expect applied researchers to fine-tune multiple versions of an LLM, so I opted for standard training hyperparameters: 200 warmup steps, batch size of 2 with 10 gradient accumulation steps, weight decay of 0.05, and an AdamW optimizer with a learning rate of  $5 \times 10^{-5}$ . I run the model for three epochs, saving the model after the each epochs. The number of epochs can affect performance, with lower epochs leading to undertraining and higher epochs leading to overfitting. By saving the model after the first and third epochs, I can compare the performance of the two models without having to retrain the model.

## D.1 Selecting generation hyperparameters for synthetic tweets

I consider 56 combinations of generation hyperparameters and generate 1,000 synthetic tweets for each set.

- epoch  $\in \{1, 3\}$
- top\_p  $\in \{0.8, 0.90, 0.95, 0.99\}$
- temperature  $\in \{0.3, 0.5, 0.7, 1, 1.3, 1.5, 1.8\}$
- top\_k  $\in \{50\}$  (keep fixed)

For each batch of synthetic tweets, I sample an additional 1,000 real tweets and split the corpus into a training set (75%) and an evaluation set (25%). The classifier model is a sentence transformer model, specifically `all-MiniLM-L12-v2` (Reimers and Gurevych 2019), to represent each tweet as a 384-dimensional vector, with a logistic regression classifier on top of the embeddings. This model was chosen because it is within 1 percentage point of the accuracy of the current best transformer model, but is around 20% of the best model’s size.

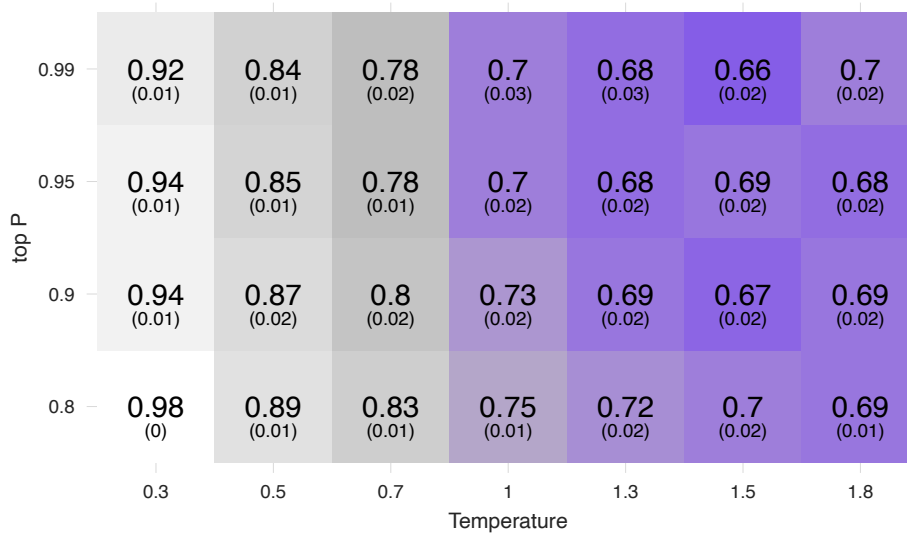
Figure SI 6 shows the ability of a classifier to distinguish between real tweets and the synthetic tweets generated from each set of hyperparameters. Lower accuracy implies better synthetic tweet quality, as the model has a harder time distinguishing real and synthetic tweets.

## D.2 Sample of generated tweets

The first ten tweets generated by a adapted/fine-tuned GPT-2 model using the optimal generation hyperparameters obtained from the classifier described above: top\_p= 0.90, top\_k= 50, temperature= 1.5, training epochs=3.

- 
1. [SYNTH] Yes, this is the USS Abraham Lincoln (DDG 71), which was in the Mediterranean a few weeks ago. And yes, you absolutely need to have a GoFund Ukraine account to donate money to charity organizations, including The Red Cross.
  2. [SYNTH] An Osint Bunker article detailing the recent Russian buildup near Kyiv  
You can click on the link to be taken to the actual map
  3. [SYNTH] Russian military is getting massed in Belarus, most likely to set up new staging areas.
  4. [SYNTH] The system is relatively good at engaging low/medium armored targets, like BTRs, MT-LBs, APCs and SPGs
  5. [SYNTH] I think people got the wrong impression from today’s press conference, where Lukashenko said “I do not fear Western military threats but Russia is prepared to pay a heavy price for any military action.
  6. [SYNTH] And as usual people are falling for the bait and trying to equate this to some sort of new high crime, especially when we see examples of this already in the news.  
We already saw it yesterday with @Nrg8000
  7. [SYNTH] This is mostly because air defence is weak, and even non TB2s could get shot down. Only a very few aircraft flew today, with the majority of them from the western part of Ukraine. In the north of Ukraine a lack of TB2s has caused large losses. The Ukrainians are probably using the drones to spot artillery strikes.
  8. [SYNTH] Tanks on the other side of the Irpin River
  9. [SYNTH] Russian forces pushed back from Kharkiv tonight  
Kherson Oblast
  10. [SYNTH] It doesn’t even have infrared sensors - only a SINCARS system.  
This basically tells you what its main purpose is.

**Table SI 7:** *Synthetically generated tweets from a GPT-2 model adapted/fine-tuned on 20,000 tweets reporting open source intelligence on the war in Ukraine. Due to Twitter’s restrictions on including actual tweets in published work, no comparison is provided for real tweets. GPT-2 generation parameters: top\_p= 0.90, top\_k= 50, temperature= 1.5, fine-tuning epochs=3*



**Figure SI 6:** *Effect of GPT-2 generation hyperparameters on synthetic tweet quality. Accuracy is the mean out-of-sample accuracy of an embedding+logistic regression model to discriminate between real and synthetic tweets. Lower accuracy implies better synthetic tweet quality, as the model has a harder time distinguishing real and synthetic tweets. Numbers in parentheses report standard deviation across 10 runs (varying the sampled real tweets and the train/test split).*

### D.2.1 GPT-3.5 and GPT-4

GPT-3.5 and GPT-4 both perform much better than GPT-2 in most tasks. However, their performance in generating tweets related to the post-2022 phase of the Russia-Ukraine war is much worse than GPT-2 than the fine-tuned GPT-2 discussed above. This is the result of the limitations of the training data of both GPT-3.5 and GPT-4, which did not include the post-2022 phase of the war at the time of writing.

I prompt GPT-3.5 with the prompt below, providing context about the war in Ukraine and providing a specific and well-known open source intelligence analyst who covers the war. Note that most of the tweets generated by GPT-3.5 discuss the pre-2022 phase of the conflict.

You're @markito0171, an OSINT analyst who covers the war in Ukraine.

Write 5 of their tweets reporting details on the war.

1. [SYNTH] “ BREAKING: Intense clashes reported in eastern Ukraine near the town of Mariupol. Local sources suggest heavy artillery fire and ground skirmishes. Monitoring the situation closely for updates. #Ukraine #WarUpdates #OSINT”
2. [SYNTH] “ UPDATE: Satellite imagery reveals a significant build-up of Russian military equipment near the border with Ukraine. The deployment includes tanks, artillery, and air defense systems. Tensions rise as the potential for further escalation increases. #Russia #Ukraine #OSINT”
3. [SYNTH] “ ANALYSIS: Recent drone footage confirms ongoing construction of trenches and fortifications by Ukrainian armed forces in the town of Avdiivka. These defensive measures indicate an anticipation of further hostilities in the region. #Ukraine #WarAnalysis #OSINT”

Repeating the exercise with GPT-4 (in July 2023) provides more fluent tweets, but are still limited by the lack of training data on the post-2022 phase of the war.

You’re @markito0171, an OSINT analyst who covers the war in Ukraine.

Write 5 of their tweets reporting details on the war.

1. [SYNTH] “#ConflictUpdate: Reports of increased shelling near Mariupol, indicating potential escalation by separatist forces. Residents are urged to stay vigilant and take necessary precautions. #UkraineConflict #OSINT”
2. [SYNTH] “#WarInUkraine: Recent satellite images reveal a build-up of heavy weaponry in Donetsk region. This suggests a significant escalation in the conflict. Please stay safe. #OSINT #Ukraine”
3. [SYNTH] “Recent OSINT evidence suggests a surge in cyber-attacks against Ukrainian govt websites. These acts of digital warfare imply an intensification of the conflict. Everyone, please keep your online data secure. #CyberWarfare #UkraineConflict”

### D.3 LLM pretraining data provenance

The training corpora used for GPT-3, 3.5, and 4 are unknown. GPT-2 was pretrained on a corpus of text described in Radford et al. (2019). The corpus was obtained by scraping all pages linked to from Reddit with at least three upvotes. However, the corpus itself has not been released.

## E Supplemental Information References

Miller, B., F. Linder, and W. R. Mebane. 2020. “Active Learning Approaches for Labeling Text: Review and Assessment of the Performance of Active Learning Approaches.” *Political Analysis*, 1–20.

Morris, J., E. Lifland, J. Yong Yoo, J. Grigsby, D. Jin, and Y. Qi. 2020. “TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 119–26.

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. 2019. “Language models are unsupervised multitask learners”.